Instructors:

Dr. David Babcock (12:30 KEC119)  
Office Hours: M 1:00-3:00; T/Th 10:00-12:00  
101 Kinsley Engineering Center  
815-6442  
email: dbabcock@ycp.edu  
webpage: http://faculty.ycp.edu/~dbabcock

Dr. Laura Garrison (9:30 KEC 130)  
Office Hours: MF 9:00 – 10:00; T 1:30-3:15;  
Th 8:30-9:30  
107 Kinsley Engineering Center  
815-6427  
email: lgarrison@ycp.edu  
webpage: http://faculty.ycp.edu/~lgarrison

Dr. David Hovemeyer (9:30 KEC 119)  
Office Hours: MWF 10-11 AM; T/Th 2-3 PM  
113 Kinsley Engineering Center  
815-6582  
email: dhovemey@ycp.edu  
webpage: http://faculty.ycp.edu/~dhovemey

Dr. Stephen Kuchnicki (12:30 KEC 130)  
Office Hours: M 11-12; W 9-11; F 8:30-10:30  
104 Kinsley Engineering Center  
815-1547  
email: skuchnic@ycp.edu  
webpage: http://faculty.ycp.edu/~skuchnic

Dr. Kala Meah (9:30 and 12:30 KEC 123)  
Office Hours: M 8:00 – 10:00; T 8:00-9:00;  
W 1:00-3:00  
111 Kinsley Engineering Center  
815-1238  
email: kmeah@ycp.edu  
webpage: http://faculty.ycp.edu/~kmeah

Required Text:


Prerequisites: None.

**CATALOG DESCRIPTION:** This course introduces the fundamental techniques of algorithm design and program construction using procedural constructs. Topics will include problem analysis; algorithm design; and implementation and debugging strategies using good programming practices. The course will cover basic data structures including variables, arrays, strings, records, and pointers; and control structures including decisions, iterations, functions, and file I/O. The course will focus on applications from computer science and engineering using C/C++. 
**COURSE OUTCOMES:** The basic objectives of this course are to provide the student with enough programming tools and methods to feel comfortable writing C / C++ / C# programs that solve problems encountered in computer science and engineering. The following topics will tentatively be covered:

- **CO1:** Be able to create, compile, and run C programs using a Gnu C/C++ compiler
- **CO2:** Be able to declare variables of various types and write programs that do basic arithmetic operations
- **CO3:** Understand and apply control structures such as conditionals and loops
- **CO4:** Understand and use arrays and vectors
- **CO5:** Understand and be able to write functions
- **CO6:** Understand and be able to define and use structure types
- **CO7:** Write programs that write to and read from files
- **CO8:** Understand and be able to use character strings
- **CO9:** Understand and be able to use pointers
- **CO10:** Use C# and Visual Studio to create basic Graphical User Interfaces.
- **CO11:** Use C# to produce graphics other than form components
- **CO12:** Use top-down design to decompose complex problems into simpler problems

**GRADING:**

- Homework (weekly programming projects) 30%
- Four Midterm Exams (10% each) 40%
- Labs/Attendance/Class Participation 10%
- Final Exam 20%

**GRADING SCALE:** The grading scale for the course is given below. Please note that this grading scale represents the most stringent criteria the students will have to meet to get a particular grade. Each instructor reserves the right to reduce (i.e. curve) the grade scale based on the final course performance.

<table>
<thead>
<tr>
<th>Grade</th>
<th>Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.0 (A)</td>
<td>≥ 90 and &lt; 100</td>
</tr>
<tr>
<td>3.5 (B+)</td>
<td>≥ 87 and &lt; 90</td>
</tr>
<tr>
<td>3.0 (B)</td>
<td>≥ 80 and &lt; 87</td>
</tr>
<tr>
<td>2.5 (C+)</td>
<td>≥ 77 and &lt; 80</td>
</tr>
<tr>
<td>2.0 (C)</td>
<td>≥ 70 and &lt; 77</td>
</tr>
<tr>
<td>1 (D)</td>
<td>≥ 60 and &lt; 70</td>
</tr>
<tr>
<td>0 (F)</td>
<td>&lt; 60</td>
</tr>
</tbody>
</table>

**PARTICIPATION AND ATTENDANCE:** Do not miss class! Since the classes are 2 hours long, you will fall far behind by missing even one. Also, note that 10% of your grade is based on your in-class lab work, attendance, and participation. If you have an emergency and cannot attend your section, you may come to another section with permission from the instructor of that section. Always notify the instructor before the class if you have to miss class. It is the student’s responsibility to get notes, announcements, and homework assignments from other students or the instructor if a class is missed. An absence will be excused only with written proof of an illness or other emergency.

**IN-CLASS LABORATORIES:** Students will work on programs during most class periods. At the end of the class, these will be submitted electronically. They will be graded on a pass/fail basis based mainly on effort.

**HOMEWORK POLICY:** Homework problems (programming assignments) will be assigned periodically. You can ask each other questions and brainstorm possible solution techniques together, but **all of the code that you write must be your own!** Since copying is especially problematic in computer programming courses, any incidences of copying will result in zeros for everyone involved and referral to the Academic Dean (see section entitled “Academic Integrity”, below). Copying includes, but is not limited to, copying from other students (including work completed in a previous semester), web sites, books, etc. To receive credit for a programming assignment, you will need to submit a signed attestation confirming that the work you submitted is your own.
Ten percent will be deducted from the homework grade for each day it is late up to three days (not three class periods) late. Homework more than three days late will receive a zero.

**EXAMS:** Exams will be closed-book, closed-notes. They will last approximately 50 minutes and will be administered at the beginning of the class period. The last half of the class will be used for instruction and/or laboratory work.

**OFFICE HOURS:** Assistance is available from the instructor during posted office hours, or by appointment. If the posted hours are in conflict with your schedule, feel free to schedule an appointment to accommodate you.

**COMMUNICATION STANDARDS:** York College recognizes the importance of effective communication in all disciplines and careers. Therefore, students are expected to competently analyze, synthesize, organize, and articulate course material in papers, examinations and presentations. In addition, students should know and use communication skills current to their field of study, recognize the need for revision as part of their writing process, and employ standard conventions of English usage in both writing and speaking. Students may be asked to further revise assignments that do not demonstrate effective use of these communication skills.

**USE OF PERSONAL TECHNOLOGY IN THE CLASSROOM:** While York College recognizes students’ need for educational and emergency-related technological devices such as laptops, PDA’s, cellular phones, etc., using them unethically or recreationally during class time is never appropriate. The college recognizes and supports faculty members’ authority to regulate in their classrooms student use of all electronic devices.

**ACADEMIC INTEGRITY POLICY:** Academic dishonesty will not be tolerated at York College. Academic dishonesty refers to actions such as, but not limited to, cheating, plagiarism, fabricating research, falsifying academic documents, etc., and included all situations where students make use of the work of others and claim such work as their own.

When an instructor believes that a student has committed an act of academic dishonesty, the instructor must provide written notification to the student, the Department Chair, and the Dean of Academic Affairs of the charge and the sanction. Documentation related to instances of academic dishonesty will be kept on file in the student’s permanent record. If the academic dishonesty is the student’s first offense, the instructor will have the discretion to decide on a suitable sanction up to a grade of 0 for the course. Students are not permitted to withdraw from a course in which they have been accused of academic dishonesty.

Students who believe they have been unjustly charged or sanctioned in cases involving a first offense must discuss the situation with the instructor immediately. Following this discussion, students may request through the Dean of Academic Affairs that the Student Welfare Committee conduct a hearing to review the charge and/or the sanction in the case. In cases of a first offense, the instructor may request that the Student Welfare Committee conduct a hearing and decide on the sanction, which can involve academic suspension or dismissal from the College, if the instructor believes the offense of be of an extremely egregious nature.

If the Dean of Academic Affairs determines that the academic dishonesty is the student’s second offense, the Dean will provide written notification to the student, the instructor, and the Department Chair. The Student Welfare Committee will automatically conduct a hearing to review the charge and decide on an appropriate sanction, which will involve academic suspension or dismissal from the College. Students who believe the Student Welfare Committee has unjustly sanctioned them may submit a written request to the Dean of Academic Affairs for a review of their case by the Dean.

**DISCLAIMER:** This syllabus is subject to revision by the instructors.
**SCHEDULE:** The following schedule is tentative, and may be subject to minor revisions.

<table>
<thead>
<tr>
<th>Week</th>
<th>Note</th>
<th>Topic</th>
<th>Reading</th>
</tr>
</thead>
<tbody>
<tr>
<td>Week 1: Jan 19-23</td>
<td><strong>No class Tuesday</strong></td>
<td>Fundamentals</td>
<td>Chapters 1-2</td>
</tr>
<tr>
<td>Week 2: Jan 26-30</td>
<td></td>
<td>Variables, data types, expressions</td>
<td>Chapters 3-4</td>
</tr>
<tr>
<td>Week 3: Feb 2-6</td>
<td></td>
<td>Decisions, Loops</td>
<td>Chapters 5-6</td>
</tr>
<tr>
<td>Week 4: Feb 9-13</td>
<td></td>
<td>Loops, continued</td>
<td></td>
</tr>
<tr>
<td>Week 5: Feb 16-20</td>
<td></td>
<td>Arrays</td>
<td>Chapter 7</td>
</tr>
<tr>
<td>Week 6: Feb 23-27</td>
<td></td>
<td>Arrays, continued</td>
<td></td>
</tr>
<tr>
<td>Week 7: Mar 2-6</td>
<td><strong>Winter break: no class</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Week 8: Mar 9-13</td>
<td></td>
<td>Functions</td>
<td>Chapter 8</td>
</tr>
<tr>
<td>Week 9: Mar 16-20</td>
<td></td>
<td>Functions, continued</td>
<td></td>
</tr>
<tr>
<td>Week 10: Mar 23-27</td>
<td></td>
<td>Structures</td>
<td>Chapter 9</td>
</tr>
<tr>
<td>Week 11: Mar 30-Apr 3</td>
<td></td>
<td>Pointers</td>
<td>Chapter 11</td>
</tr>
<tr>
<td>Week 12: Apr 6-10</td>
<td><strong>No class Thursday</strong></td>
<td>Intro to C#</td>
<td></td>
</tr>
<tr>
<td>Week 13: Apr 13-17</td>
<td></td>
<td>Graphical User Interfaces (GUIs)</td>
<td></td>
</tr>
<tr>
<td>Week 14: Apr 20-24</td>
<td></td>
<td>More Graphics and GUI elements</td>
<td></td>
</tr>
<tr>
<td>Week 15: Apr 27-May 1</td>
<td></td>
<td>File I/O, Images</td>
<td></td>
</tr>
<tr>
<td>Week 16: May 4-8</td>
<td></td>
<td>Misc. C# topics</td>
<td></td>
</tr>
</tbody>
</table>